

programmed to place certain components on a particular board, in this case a mobile phone board, and through the use of a machine setup for the particular assembly line. It is evident, therefore, that the final assembled board can have different characteristics or features based not only on the type of components that are placed on it, but also on the particular machine setup that is used on the line. In other words, the over-all instructions, or recipe, to be programmed into the computer-controlled system must include a sub-recipe having a list of components, a placement list for those components and, at the same time, another sub-recipe describing the setup characteristics of the line, including the feeder characteristics, as an example for the case shown in Fig. 2a.

[0027] When programming a manufacturing line anew with a new object, it is important to note that that new object is automatically assigned the version V1.00, regardless of the versions of the sub-objects, which will be described at greater length below in the description of the embodiments. Thus, in building the mobile phone process flow shown in Fig. 2a, first the sub-objects are created in the program: namely, package form (10), i.e., the geometry of the area, including spacing, into which the transistor components will be placed; components (20) themselves; placement list (30), i.e., the places where the components will be mounted on the board; board (40). Each sub-object has a version V1.00. Similarly, the sub-objects on the mobile phone sub-recipe side of the main-recipe also are assigned version V1.00: namely, feeder (55) for feeding components in placement on the board, and

setup for the mobile phone line. Finally, the main-object, that is, main-recipe (70) for all the instructions in the sub-recipes also gets version V1.00. Even if the sub-objects are of a different version, the main object still gets version V1.00. At this point, the version is not yet released for manufacturing.

5

[0028] However, once the main recipe (70) is first created with version V1.00, any subsequent changes and modifications in the sub-objects will be reflected in the revision number for all the sub-objects upstream from that changed sub-object, up to and including the main-recipe. Thus, if a mobile phone board built with recipe V1.00 having component (20) with a package form factor (10) is defective because of an inadequate tolerance prescribed in that form factor, a different form factor can be assigned, thereby changing the version for the package form to V2.00. Any sub-object that does not comprise, and, therefore is not influenced by other sub-sub-objects, as in the case with the package form factor, will have its second set of numbers of its version number unchanged at .00. However, as pointed out above, any sub-object that is influenced by changes in other sub-objects, will have their version numbers incremented by .01. Hence the version numbers V1.01 shown in Fig. 2b.

20

[0029] Any change on the right hand side leg of Fig. 2b, that is, in the collective sub-recipe of the right hand side (80) of the same Figure, which can be characterized as board recipe, does also affect the sub-recipe on the left side (90), which also can be characterized as setup or line recipe.

Because the package form is now of version V2.00, the feeder incorporating that package form takes on version V1.01, and so does the setup sub-object. If, now, the feeder is modified for some reason, and becomes version 2.00, for example, that will affect the setup sub-object recipe, and the main recipe or object (70) will also have its version incremented by .01 to V1.02. This is shown in Fig. 2c. Thus, the main object, which is upstream of the setup recipe, shows two changes in its sub-objects. Sub-recipes from which those changes emanate should preferably be given unique names, such as "Setup" or "Line" recipe on the left side of Fig. 2b, and "Board" recipe for the right side in the same Figure so that the user can trace back where the change or modifications took place.

[0030] After the initial programming of a computer-controlled manufacturing system, any subsequent modification, that is, editing of the objects, takes on a different significance depending upon whether the edited version is released or not. First, when a version 1.00 is downloaded to the line, the machines on the line will allow only one board to run through the line, but in the "run-in" mode only. A second board is allowed to move into the input conveyor of a machine, but requires the user to press the start button before it can be assembled. The user will be asked to check the entire board and confirm if the board got assembled without any errors. To help in checking the modifications, graphical printout as well as textual printout of the modified parameters will be displayed at the station or can be printed to help in finding the modification to an earlier released version. As soon as the user